club commodore

Boletín informativo para los usuarios de microordenadores VIC y CBM



MONITOR EN COLOR C-1701



octubre 1983

ADDRESS		DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0	DESCRIPTION
00	($00)	M0X7	M0X6	M0X5	M0X4	M0X3	M0X2	M0X1	M0X0	MOB 0 X-position
01	($01)	M0Y7	M0Y6	M0Y5	M0Y4	M0Y3	M0Y2	M0Y1	M0Y0	MOB 0 Y-position
02	($02)	M1X7	M1X6	M1X5	M1X4	M1X3	M1X2	M1X1	M1X0	MOB 1 X-position
03	($03)	M1Y7	M1Y6	M1Y5	M1Y4	M1Y3	M1Y2	M1Y1	M1Y0	MOB 1 Y-position
04	($04)	M2X7	M2X6	M2X5	M2X4	M2X3	M2X2	M2X1	M2X0	MOB 2 X-position
05	($05)	M2Y7	M2Y6	M2Y5	M2Y4	M2Y3	M2Y2	M2Y1	M2Y0	MOB 2 Y-position
06	($06)	M3X7	M3X6	M3X5	M3X4	M3X3	M3X2	M3X1	M3X0	MOB 3 X-position
07	($07)	M3Y7	M3Y6	M3Y5	M3Y4	M3Y3	M3Y2	M3Y1	M3Y0	MOB 3 Y-position
08	($08)	M4X7	M4X6	M4X5	M4X4	M4X3	M4X2	M4X1	M4X0	MOB 4 X-position
09	($09)	M4Y7	M4Y6	M4Y5	M4Y4	M4Y3	M4Y2	M4Y1	M4Y0	MOB 4 Y-position
10	($0A)	M5X7	M5X6	M5X5	M5X4	M5X3	M5X2	M5X1	M5X0	MOB 5 X-position
11	($0B)	M5Y7	M5Y6	M5Y5	M5Y4	M5Y3	M5Y2	M5Y1	M5Y0	MOB 5 Y-position
12	($0C)	M6X7	M6X6	M6X5	M6X4	M6X3	M6X2	M6X1	M6X0	MOB 6 X-position
13	($0D)	M6Y7	M6Y6	M6Y5	M6Y4	M6Y3	M6Y2	M6Y1	M6Y0	MOB 6 Y-position
14	($0E)	M7X7	M7X6	M7X5	M7X4	M7X3	M7X2	M7X1	M7X0	MOB 7 X-position
15	($0F)	M7Y7	M7Y6	M7Y5	M7Y4	M7Y3	M7Y2	M7Y1	M6Y0	MOB 7 Y-position
16	($10)	M7X8	M6X8	M5X8	M4X8	M3X8	M2X8	M1X8	M0X8	MSB of X-position
17	($11)	RC8	ECM	BMM	DEN	RSEL	Y2	Y1	Y0	See text
18	($12)	RC7	RC6	RC5	RC4	RC3	RC2	RC1	RC0	Raster register
19	($13)	LPX8	LPX7	LPX6	LPX5	LPX4	LPX3	LPX2	LPX1	Light Pen X
20	($14)	LPY7	LPY6	LPY5	LPY4	LPY3	LPY2	LPY1	LPY0	Light Pen Y
21	($15)	M7E	M6E	M5E	M4E	M3E	M2E	M1E	M0E	MOB Enable
22	($16)	—	—	RES	MCM	CSEL	X2	X1	X0	See text
23	($17)	M7YE	M6YE	M5YE	M4YE	M3YE	M2YE	M1YE	M0YE	MOB Y-expand
24	($18)	VM13	VM12	VM11	VM10	CB13	CB12	CB11	—	Memory Pointers
25	($19)	IRQ	—	—	—	ILP	IMMC	IMBC	IRST	Interrupt Register
26	($1A)	—	—	—	—	ELP	EMMC	EMBC	ERST	Enable Interrupt
27	($1B)	M7DP	M6DP	M5DP	M4DP	M3DP	M2DP	M1DP	M0DP	MOB-DATA Priority
28	($1C)	M7MC	M6MC	M5MC	M4MC	M3MC	M2MC	M1MC	M0MC	MOB Multicolor Sel
29	($1D)	M7XE	M6XE	M5XE	M4XE	M3XE	M2XE	M1XE	M0XE	MOB X-expand
30	($1E)	M7M	M6M	M5M	M4M	M3M	M2M	M1M	M0M	MOB-MOB Collision
31	($1F)	M7D	M6D	M5D	M4D	M3D	M2D	M1D	M0D	MOB-DATA Collision
32	($20)	—	—	—	—	EC3	EC2	EC1	EC0	Exterior Color
33	($21)	—	—	—	—	B0C3	B0C2	B0C1	B0C0	Bkgd #0 Color
34	($22)	—	—	—	—	B1C3	B1C2	B1C1	B1C0	Bkgd #1 Color
35	($23)	—	—	—	—	B2C3	B2C2	B2C1	B2C0	Bkgd #2 Color
36	($24)	—	—	—	—	B3C3	B3C2	B3C1	B3C0	Bkgd #3 Color
37	($25)	—	—	—	—	MM03	MM02	MM01	MM00	MOB Multicolor #0
38	($26)	—	—	—	—	MM13	MM12	MM11	MM10	MOB Multicolor #1
39	($27)	—	—	—	—	M0C3	M0C2	M0C1	M0C0	MOB 0 Color
40	($28)	—	—	—	—	M1C3	M1C2	M1C1	M1C0	MOB 1 Color
41	($29)	—	—	—	—	M2C3	M2C2	M2C1	M2C0	MOB 2 Color
42	($2A)	—	—	—	—	M3C3	M3C2	M3C1	M3C0	MOB 3 Color
43	($2B)	—	—	—	—	M4C3	M4C2	M4C1	M4C0	MOB 4 Color
44	($2C)	—	—	—	—	M5C3	M5C2	M5C1	M5C0	MOB 5 Color
45	($2D)	—	—	—	—	M6C3	M6C2	M6C1	M6C0	MOB 6 Color
46	($2E)	—	—	—	—	M7C3	M7C2	M7C1	M7C0	MOB 7 Color

NOTE: A dash indicates a no connect. All no connects are read as a "1."

Depósito Legal Sep. B. 2133-1958 - Altés, s.l., Caballero 87, Barcelona

EDITORIAL

más novedades de COMMODORE

MICROELECTRÓNICA Y CONTROL S. A. anuncia la aparición de nuevos periféricos para sus Ordenadores VIC-20 y COMMODORE 64. Estos son: El "plotter" 1520, el nuevo monitor de video modelo C-1701 y la impresora1525. Veamos a continuación las características técnicas de estos nuevos componentes de la familia de ORDENADORES FAMILIARES Y PERSONALES de COMMODORE:

"PLOTTER" VIC 1520

Método de impresión: *Dibujo mediante bolígrafos de diseño especial.*

Color: *cuatro colores: negro, azul, verde y rojo con cambio desde programa.*

Cabezal: *"Plotter" X-Y tipo tambor.*

Velocidad de impresión: *Media de 14 car./seg.*

Caracteres por línea: *Máximo 80 carac., formatos de 80, 40, 20 y 10 carac./línea.*

Juego de caracteres: *96. Posibilidad de imprimir los caracteres con cuatro orientaciones distintas.*

Velocidad de dibujo: *264 pasos/seg.*

Longitud del paso: *0,2 mm en dirección X e Y.*

Velocidad de dibujo de línea: *52,8 mm/seg. en dirección X e Y. 73 mm/seg. en una línea a 45 grados.*

Área de dibujo: *480 pasos (96 mm) en dirección X. Programable en dirección Y (Máx. ± 999 de una sola vez).*

Papel: *Rollo de 4,5 pulgadas (114 mm).*

Dimensiones: *Ancho 282 mm, Altura 93 mm, Profundidad 251 mm*

MONITOR EN COLOR C-1701

Pantalla: *13 pulgadas (330 mm).*

Capacidad de representación: *25 líneas de 40 caracteres.*

Resolución: *320 líneas horizontales.*

Compatibilidad: *VIC-20 y COMMODORE 64. Conectable a un grabador de video.*

Amplificador y altavoz: *Incorporados.*

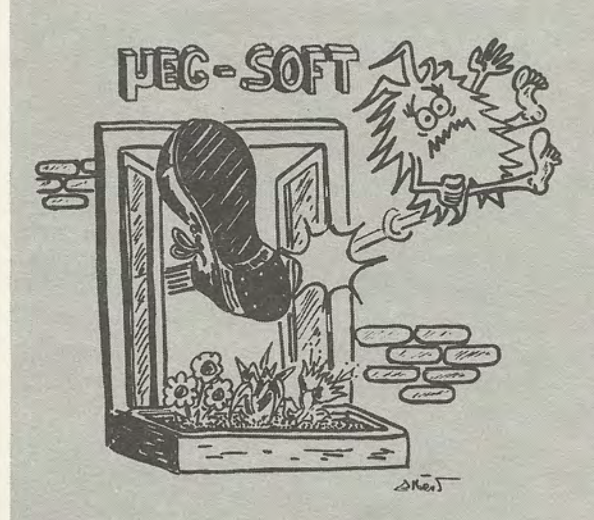
IMPRESORA VIC-1525

Método de impresión: *Impacto por matriz de puntos y martillo único.*

Matriz de carácter: *6 por 7 puntos.*

Caracteres: *Mayúsculas, minúsculas y caracteres gráficos de teclado del VIC-20 y COMMODORE 64.*

(continúa en la pág. siguiente)



VENTANA CBM

rutinas de cálculo del interpretador BASIC (y II)

por **R. NAVARRO**
(M.E.C. SOFT)

Terminamos en este número con la serie de rutinas matemáticas comenzadas en el anterior.

BASIC 1	BASIC 2	BASIC 4	DESCRIPCIÓN
C863	C873	B8F6	Busca un entero y deja su valor en (S11).
C91C	C928	B9AB	Suma dígito ASCII al acumulador # 1 (en S1F). Entregar código ASCII en .Y
CCB8	CC9F	BD98	Entrar y evaluar cualquier expresión BASIC. Puede decirse que es una de las rutinas más completas del Interpretador. Permite evaluar cualquier expresión, alfa o numérica, entera, flotante o dimensionada. El puntero de texto (CHRGET) debe apuntar al principio de la expresión. En datos de salida, actualiza el tipo de variable, el tipo de variable numérica si pro-

(continúa en la pág. siguiente)



VENTANA CBM

rutinas de cálculo del interpretador BASIC

(viene de la pág. anterior)

BASIC 1	BASIC 2	BASIC 4	DESCRIPCIÓN
			cede, el acumulador flotante # 1, si es numérica (siempre en coma flotante aunque la expresión sea entera (a%+b%). Si se trata de una expresión alfa, debe efectuarse un salto posterior a ($D57B, $D57D ó $C7B5) — descartar cadena temporal — y se obtendrá en $1f-$20, ($71-$72 en BASIC 1) un puntero al descriptor de la cadena resultante de la evaluación (1en+pb+pa).
CED6	CEC8	C086	OR entre dos enteros de 2 bytes.
CED9	CECB	C089	AND entre dos enteros de 2 bytes. Resultado en acumulador # 1
D287	D27C	C4CB	Transfiere el contenido de .Y al acumulador # 1 en coma flotante.
D275	D733	C986	Resta. Coloca en acumulador # 1, la diferencia entre el acumulador # 2 y el # 1.
D73C	D773	C99D	Suma. Coloca la suma de los acumuladores # 1 y # 2 en el acumulador # 1.
D8FD	D934	CB5E	Multiplicación.
D95E	D998	CBC2	Carga de los 5 bytes de memoria apuntados por .A (peso bajo) e .Y (peso alto) sobre el acumulador # 2. Útil para transferencia de contenidos de variables internas al acumulador # 2 y operar después entre acumuladores.
D9E1	DA1B	CC45	División. Carga en el acumulador # 1, el resultado de dividir el acumulador # 2 por el # 1.
DA74	DAAE	CCD8	Carga el acumulador # 1 con los cinco bytes apuntados por .A e .Y.
DA99	DAD3	CCFD	Transfiere el contenido del acumulador # 1 al punto de memoria apuntado por .X (bajo) e .Y (peso alto). En este proceso se incluye un redondeo (no pasar a entero) de los 6 bytes de capacidad de los acumuladores a los 5 de que disponen las variables.
DACE	DB08	CD32	Trasladar el contenido del acumulador # 2 al acumulador # 1.
DBC5	DBFF	CE29	Convierte el contenido del acumulador # 1 en una cadena que coloca a partir del punto $1000 de la memoria. Al final de la cadena coloca un cero binario, ya que no se conoce su longitud. ■

EDITORIAL

más novedades de COMMODORE

(viene de la pág. anterior)

Gráficos: *Punto a punto, 7 puntos verticales y 480 horizontales por línea.*

Código de carácter: *8 bits del VIC-20 y COMMODORE 64.*

Tamaño del carácter: *Altura: 7 puntos (2,82 mm). Anchura: 6 puntos (2,53 mm).*

Velocidad de impresión: *30 caracteres por segundo. Impresión unidireccional de izquierda a derecha.*

Caracteres por línea: *80 máximo.*

Espaciado de caracteres: *10 caracteres por pulgada.*

Espaciado de líneas: *6 líneas por pulgada, Modo normal. 9 líneas por pulgada, Modo gráfico.*

Velocidad de avance de papel: *5 líneas por segundo, Modo normal. 7,5 líneas por segundo, Modo gráfico.*

Arrastre de papel: *Por tractor. Papel con orificios.*

Ancho de papel: *4,5 a 10 pulgadas (114,3 a 254 mm), incluyendo agujeros de arrastre. 8,5 pulgadas (215,9 mm) después de eliminar las bandas de arrastre.*

Número de copias: *Original y 1 ó 2 copias (dependiendo del grueso del papel).*

Cinta tintadora: *Un solo color, tipo cartucho.*

Dimensiones externas: *Profundidad: 234,5 mm. Anchura: 420 mm. Altura: 163 mm.*

Peso: *Aproximadamente 4,5 Kg.*

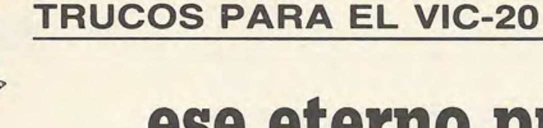
Alimentación: *220 a 240 V. ± 10%, corriente alterna de 50/60 Hz.*

Consumo de corriente: *20 vatios máximo (en impresión). 8 vatios (en reposo).*

Temperatura de trabajo: *De 5 a 40 grados centígrados.*

Humedad ambiente: *Entre 20 y 80%, sin condensación.*

Como puede verse, se trata de una buena colección de periféricos para el VIC-20 y el COMMODORE 64 algunos de ellos completamente nuevos en el mercado de los ORDENADORES PERSONALES. ■



TRUCOS PARA EL VIC-20

ese eterno problema de las memorias de video y los cartuchos de ampliación

por **P. MASATS**

Una de las peculiaridades que más problemas causan al usuario del VIC-20 es el hecho de que las memorias de pantalla y de color cambian de sitio al ampliar la memoria a más de 8K. Es importante no olvidar que si se tiene conectado el cartucho de ampliación de 3K el total de memoria es 8K, con lo que la matriz de pantalla y la de color continúan estando en el mismo sitio. A menudo necesitamos que nuestros programas se puedan adaptar automáticamente a cualquier configuración de memoria. Para ello se debe leer una posición de memoria que nos da las direcciones en que se hallan estas matrices. El procedimiento es el siguiente:

```
10 MP=PEEK(648)*256
20 MC=((PEEK(648)AND3)OR
   148)*256
```

Estas dos líneas, al principio de nuestros programas, nos asignan a las variables MP (Memoria de Pantalla) el valor de la dirección del primer byte de esta matriz y a MC (Memoria de Color) lo mismo para la matriz de color. Luego, al hacer los POKEs correspondientes, los haremos sobre estas variables en vez de sobre cantidades. Por ejemplo:

```
1000 POKE MP + (posición del
     carácter), (código del carácter)
1010 POKE MC + (posición del
     carácter), (código del color)
```

Este procedimiento hace posible que nuestros programas se adapten automáticamente a la configuración de memoria existente en el equipo en el momento de ejecutar el programa. (Ver también el artículo «UNA OJEADA AL VIC PROPIAMENTE DICHO», publicado en CLUB COMMODORE número 1, págs. 4 a 6.)

Existe otro sistema para resolver este problema, aunque es menos elegante que el anterior, pero vamos a describirlo porque — sobre todo — es ilustrativo de la manera de trabajar del equipo y no requiere ninguna modificación del programa a ejecutar. Consiste en «obligar» al VIC a convertirse en un ordenador con menos memoria sin apagarlo y desconectar el cartucho de ampliación. Se trata de hacer unos POKEs (¡cómo no!) seguidos de un SYS (para variar) que nos dejarán el equipo con una configuración de memoria determinada. El procedimiento es:

```
POKE641,0:POKE642,X:POKE643,0:
POKE644,Y:POKE648,Z:SYS64824
```

Los valores de X, Y y Z se dan en la tabla siguiente:

CONFIGURAR A	642,X	644,Y	648,Z
SIN EXPANSIÓN	16	30	30
CON 3K	4	30	30
CON 8K	18	64	16
CON 16K	18	96	16
CON 24K	18	128	16

■

cómo utilizar la palanca de juegos desde un programa en BASIC

Hemos recibido algunas consultas referentes al uso de la palanca de juegos (joystick) en el VIC-20 en programas propios. Si entramos en el VIC el programa cuyo listado se da a continuación, veremos que en la pantalla aparecen los valores de las posiciones de memoria asociadas al funcionamiento de este accesorio de juegos y la posición en que está cuando lo hacemos funcionar (no vale ponerlo al revés). Para los que necesiten más información sobre la manera de leer el estado de los interruptores que indican la posición de la palanca, éstos corresponden a los bits de las posiciones de memoria siguientes:

37137 - $9111
Bit 2 (valor 4); interruptor 0- Palanca hacia arriba.
Bit 3 (valor 8); interruptor 1 - Palanca hacia abajo.
Bit 4 (valor 16); interruptor 2 - Palanca hacia la izquierda.
Bit 5 (valor 32); Botón de disparo.
37152 - $9120
Bit 7 (valor 128); interruptor 3 - Palanca hacia la derecha.

LISTADO

```
PROGRAMA:09 JOYSTICK

10 POKE37151,0:PA=37151:PB=37152:RB
=37154

20 A= PEEK(PA):POKERB,127:B=PEEK(PB
):POKERB,255

30 PRINT"[CRSRU]"A,B

100 IF(AAND4)=0THENPRINT"ARRIBA
[CRSRU]"

110 IF(AAND8)=0THENPRINT"ABAJO
[CRSRU]"

120 IF(BAND128)=0THENPRINT"DERECHA
[CRSRU]"

130 IF(AAND16)=0THENPRINT"IZQUIERDA
[CRSRU]"

140 IF(AAND32)=0THENPRINT"BOTON
[CRSRU]"

150 GOTO20

READY.
```

■

CLUBS DE USUARIOS

editor de "sprites" para el C-64

por **JAUME JULIÁ**
(del Club de Usuarios de Ordenadores Commodore

El CBM-64 cuenta entre sus características más destacables la de poder controlar «Sprites». Un «Sprite» es un pequeño dibujo realizado por el programador, que puede moverse por la pantalla con una facilidad asombrosa, sin interferir para nada con el texto o dibujo que se encuentre debajo. Las posibilidades de los «Sprites» son enormes: pueden pasar unos por encima de otros, detectar las colisiones que se produzcan entre ellos o entre los «Sprites» y datos, expandirse en ambos sentidos, etc. Su utilidad en juegos y programas educativos está totalmente demostrada. Sin embargo, una de las tareas más engorrosas que se deben realizar para poder tener «Sprites» en un programa es diseñarlos. El presente programa sirve para confeccionar «Sprites» con suma facilidad. Aunque no es tan sofisticado como los que se desarrollan con vistas a su comercialización, es más que suficiente para que podáis dibujar «Sprites» cómodamente.

Una vez entrado el programa y antes de ejecutarlo, almacénalo en el cassette o la unidad de disco. Si todo ha ido bien, al hacer RUN la pantalla se borrará y aparecerá un mensaje preguntando por el número de inicio de las líneas DATA del primer «Sprite». La respuesta al INPUT aparece también en pantalla por lo que es suficiente que pulses RETURN.

La pantalla de edición — que aparecerá una vez contestada la pregunta — muestra a la derecha una trama de 24 × 21 puntos, con un cursor arriba a la izquierda. El cursor se desplaza por toda la trama mediante las teclas de control del cursor. Estas teclas realizan la misma función que tienen cuando editas un programa. A la izquierda de la pantalla se muestran las funciones de que dispone el programa y las teclas que debes pulsar para que las realice. Coloca el cursor en el punto que desees y pulsando la tecla de función F1 el punto se transformará en un círculo. Al mismo tiempo, abajo a la izquierda, aparecerá un pequeño puntito. En este lugar se muestra el «Sprite» tal como quedará cuando lo utilices en algún programa por lo que te será muy fácil comprobar los resultados de tu dibujo. Si te equivocas, pulsando la tecla F3 podrás borrar los puntos que desees. Las teclas F5 y F7 sirven para expandir el «Sprite» mostrado en la esquina inferior izquierda en sentido horizontal y vertical, respectivamente. Pulsando la tecla F2 cambiará el color de la pantalla, y pulsando F4 el color del Sprite».

Nos quedan dos teclas por estudiar. En caso de que el dibujo que estés haciendo no te guste, y quieras empezar de nuevo, pulsa F6. El ordenador te pedirá confirmación. Si a esta petición respondes con una «N» todo seguirá igual. Si contestas «S» el ordenador te mostrará un mensaje indicándote la respuesta que debes dar a la pregunta sobre las líneas DATA que te hará a continuación. Después de este pequeño «protocolo», la trama y el «Sprite» aparecerán completamente limpios.

Cuando estés satisfecho del «Sprite» realizado, pulsa F8, con lo que el ordenador — después de pedir confirmación — generará las líneas DATA que componen el «Sprite» y las incorporará automáticamente al programa. Cuando hayas definido todos los «Sprites» que precises, responde con un número negativo a la pregunta sobre líneas DATA. Automáticamente se borrará el programa quedando únicamente las líneas DATA que contienen las definiciones de los «Sprites», colocadas por orden. Incorpora estas líneas a tu programa y utilízalas como se indica en el manual del usuario.

ALGUNAS CONSIDERACIONES SOBRE EL PROGRAMA

Entra el programa tal como se indica en el listado, RESPETANDO INCLUSO LOS ESPACIOS. La línea número 1, la 1010 y la 1020 son muy importantes. Verifícalas concienzudamente.

Respeta la numeración de la primera línea. La línea número 1 debe ser la número 1, no la 5, la 10 ó la 100. La segunda línea — que en el listado figura como 1010 — no debe tener un número inferior a 1000. (Por supuesto, las líneas posteriores deberán tener un número mayor.)

Los valores que se dan como respuesta a los INPUT son los más ade-



cuados. No los cambies. Cuando pidas una nueva pantalla, el ordenador te dirá el número de línea que debes entrar. Respétalo o podrías estropear el programa. Además, cuando te pide que pulses ESPACIO, pulsa únicamente la barra espaciadora. La pulsación de otra tecla podría estropear el programa. Sobre todo no pulses RETURN.

El presente programa te permite crear las líneas DATA de 18 «Sprites». No quieras definir más de una vez, ya que el programa se estropearía.

Hay dos líneas MUY ÚTILES como técnica de programación, y que sirven tanto para el C-64 como para el VIC-20. Son las 1010 y 1020.

Las dos líneas mencionadas —cuando se ejecutan— se borran automáticamente, borrando también todas las líneas que las siguen. La memoria ocupada por esta parte de programa se recupera para variables. Esta técnica es muy útil sobre todo en el VIC-20, que no dispone de mucha memoria.

En próximos números hablaremos de cómo encadenar las líneas DATA a tu programa. Por último, si tienes problemas a la hora de entrar el programa o te da pereza hacerlo, el Club de Usuarios te puede facilitar una copia del mismo. Envía tu nombre y dirección junto con un talón o giro postal por 500 pesetas (cassette) o 900 pesetas (disquette) a: CLUB DE USUARIOS DE ORDENADORES COMMODORE. Vía Augusta, 120. Barcelona-6. En caso de que desees afiliarte al Club (información en la Revista número 9), éste será uno de los varios programas que te serán enviados en cassette como regalo de bienvenida. ■

LISTADO

```
1 PRINT"[ CLR ]LINEA DE INICIO DATA
  100[CRSRL][CRSRL][CRSRL][CRSRL][CRSRL]
";:INPUTLN:IFLN>9THEN1040

1010 DE=PEEK(61)+256*PEEK(62)+3:POK
E252,INT(DE/256):POKE251,DE-256*PEE
K(252)

1020 POKEDE-2,0:POKEDE-1,0:POKE45,P
EEK(251):POKE46,PEEK(252)

1030 POKE198,3:POKE631,147:POKE632,
49:POKE633,13:END

1040 H3=(LN+50)/10:H1$=STR$(H3):H1=
VAL(MID$(H1$,2,1))+48

1050 H2=VAL(MID$(H1$,3,1))+48

1060 POKE53281,14:DIMA(504),B(64):V
=53248:PV=1079:B=PV

1070 FORT=832TO896:POKET,0:NEXT

1080 POKEV+21,1:POKE2040,13:POKEV+3
9,0:POKEV,65:POKEV+1,205

1090 POKEV+23,0:POKEV+29,0

1100 PRINT"[ CLR ][ BLU ]
EDITOR  DE  SPRITES"

1110 FORT=1TO21:PRINTTAB(15)"......
..................":NEXT

1120 PRINT"[HOME ][CRSRD][CRSRD]  [RVSON]
F1[RVSOF] DIBUJAR"

1130 PRINT"[CRSRD]  [RVSON]F3[RVSOF]
 BORRAR"

1140 PRINT"[CRSRD]  [RVSON]F5[RVSOF]
 EXP. X"

1150 PRINT"[CRSRD]  [RVSON]F7[RVSOF]
 EXP. Y"

1160 PRINT"[CRSRD]  [RVSON]F2[RVSOF]
 COLOR P."

1170 PRINT"[CRSRD]  [RVSON]F4[RVSOF]
 COLOR S."

1180 PRINT"[CRSRD]  [RVSON]F6[RVSOF]
 NUEVA P."

1190 PRINT"[CRSRD]  [RVSON]F8[RVSOF
 DATA"

1200 POKEPV+PH,PEEK(PV+PH)+128

1210 GETA$:IFA$=""THEN1210

1220 IFA$="[CRSRD]"THENS=PV+PH:PV=P
V+40:B=B+16:I=1:IFPV>1879THENPV=107
9:B=PV

1230 IFA$="[CRSRU]"THENS=PV+PH:PV=P
V-40:B=B-16:I=1:IFPV<1079THENPV=187
9:B=1399

1240 IFA$="[CRSRR]"THENS=PV+PH:PH=P
H+1:I=1:IFPH>23THENPH=1

1250 IFA$="[CRSRL]"THENS=PV+PH:PH=P
H-1:I=1:IFPH<0THENPH=23

1260 IFI=1THENPOKEPV+PH,PEEK(PV+PH)
+128:POKES,PEEK(S)-128:I=0:GOTO1210

1270 IFA$="[  F1 ]"THENPOKEPV+PH,20
9:A(PV+PH-B)=1:GOTO1360

1280 IFA$="[  F3 ]"THENPOKEPV+PH,17
4:A(PV+PH-B)=0:GOTO1360

1290 IFA$="[  F5 ]"THENPOKEV+29,ABS
(PEEK(V+29)-1)

1300 IFA$="[  F7 ]"THENPOKEV+23,ABS
(PEEK(V+23)-1)

1310 IFA$="[  F8 ]"THENGOSUB1520:IF
OK$="S"THEN1420

1320 IFA$="[  F2 ]"THENPC=33:GOTO15
00

1330 IFA$="[  F4 ]"THENPC=39:GOTO15
00

1340 IFA$="[  F6 ]"THENGOSUB1520:IF
OK$="S"THEN1550

1350 GOTO1210

1360 PE=INT((PV+PH-B)/8)*8

1370 VB=128:FORT=PETOPE+7

1380 PO=PO+A(T)*VB:VB=VB/2

1390 NEXT

1400 POKE832+PE/8,PO:B(PE/8+1)=PO:P
O=0

1410 GOTO1210

1420 PRINT"[ CLR ][ BLU ]";:IN=1:FO
RT=1TO4:PRINTLN;"DATA";:LN=LN+10

1430 FORK=1TO16:B$=STR$(B(IN)):LE=L
EN(B$)-1:IN=IN+1

1440 B$=RIGHT$(B$,LE)

1450 PRINTB$;",";:NEXT

1460 PRINT"[CRSRL] ":NEXT

1470 POKE198,10:POKE631,19

1480 FORT=632TO635:POKET,13:NEXT

1490 POKE636,82:POKE637,213:POKE638
,13:POKE639,H1:POKE640,H2:END

1500 CO=PEEK(V+PC)AND15:CO=CO+1:IFC
O>15THENCO=0

1510 POKEV+PC,CO:GOTO1210

1520 PRINT"[HOME ][CRSRD][CRSRD][CRSRD]
[CRSRD][CRSRD][CRSRD][CRSRD][CRSRD]
[CRSRD][CRSRD][CRSRD][CRSRD][CRSRD]
[CRSRD][CRSRD][CRSRD][CRSRD][CRSRD]
[CRSRD][CRSRD][CRSRD][CRSRD][CRSRD]
"TAB(18)"CONFIRME POR FAVOR"

1530 GETOK$:IFOK$=""THEN1530

1540 FORT=1962TO1979:POKET,32:NEXT:
RETURN

1550 PRINT"[ CLR ]"TAB(15)"[RVSON]A
TENCION[RVSOF]"

1560 PRINT"[CRSRD][CRSRD]   UTILICE
 COMO PROXIMA LINEA DE INICIO    DA
TA LA"LN

1570 PRINT"[CRSRD][CRSRD]PULSE   [RVSON]
   ESPACIO   [RVSOF]"

1580 POKE198,0:WAIT198,1:RUN

READY.
```



SOFTWARE DE BASE (VII)

rutina de búsqueda dicotómica

por **E. MARTÍNEZ DE CARVAJAL**

En los primeros artículos de esta sección, SOFTWARE DE BASE, os hablé de diferentes métodos para ordenar tablas de elementos. Una de las principales aplicaciones de la ordenación de tablas es la de facilitar la localización de los elementos dentro de ellas. Es evidente que es mucho más fácil buscar, por ejemplo, el nombre de una persona en una lista si está ordenada alfabéticamente. ¿Quién no ha tenido que buscar, a veces sin éxito, su nombre en una lista de las personas aprobadas en un examen? Veamos cuál es el proceso de búsqueda que seguimos:

Primero miramos un elemento más o menos intermedio de la lista.

Si su nombre es menor que el nuestro damos un salto hacia adelante en la tabla, dependiendo del grado de diferencia.

Si el nombre es mayor, el salto es hacia atrás, también dependiendo del grado de diferencia.

Este proceso lo repetimos hasta que, o encontramos nuestro nombre o, lo que es más corriente, vemos que tendría que estar entre dos consecutivos y no está.

El manejo de tablas en un ordenador también se facilita si están ordenadas. Y se puede aplicar este mismo procedimiento de búsqueda para que el ordenador localice un elemento dentro de una tabla ordenada, diciéndonos el número de elemento de que se trata. Es la base del método conocido con el nombre de búsqueda dicotómica o binaria. En este método no se contempla el concepto de «salto hacia adelante o atrás dependiendo del grado de diferencia», ya que si bien se podría hacer, la diferencia de velocidad en la búsqueda no merece la pena y menos a nuestros niveles en los que un tiempo máximo de acceso de menos de un segundo en una tabla de 100 elementos está más que bien. ¿O no? En la rutina, el punto inicial de búsqueda es el punto medio de la tabla. Si el nombre que se busca es menor, este elemento pasa a ser el último de la lista, y si es mayor, pasa a ser el primero, repitiéndose el proceso hasta que, o bien se encuentra, o bien resulta que el primer y el último son dos elementos consecutivos. De una manera más gráfica lo que se hace es ir partiendo la tabla por la mitad una y otra vez, por lo que la convergencia hacia el punto donde debería estar el elemento buscado es muy rápida. La rutina se muestra en el listado 1.

Líneas 1000-1090: Ejemplo de utilización de la rutina de búsqueda dicotómica para la localización de una palabra de dos letras dentro de una tabla ordenada de 100 elementos.

LISTADO 1

```
1000 REM *** EJEMPLO DE UTILIZACION
 DE BUSQUEDA DICOTOMICA ***
1010 REM ---------------------------
---------------------------
1020 REM E. MTNZ. DE CARVAJAL HEDRI
CH
1030 REM 2/9/1983
1040 REM ---------------------------
---------------------------
1045 GOSUB 1320
1050 INPUT "NOMBRE : ";N$
1060 L2=100
1070 GOSUB 1110:REM DICOTOMICA
1080 PRINT"ES EL ELEMENTO :";X
1090 GOTO 1050
1100 REM ---------------------------
---------------------------
1110 REM *** BUSQUEDA DICOTOMICA EN
 UNA TABLA ORDENADA ***
1120 REM
1130 REM DATOS DE ENTRADA
1140 REM
1145 REM T$() --> TABLA OREDENADA D
E ELEMENTOS
1150 REM N$   --> NOMBRE A BUSCAR
1160 REM L2   --> NUM. ELEMENTOS DE
 LA TABLA
1170 REM
1180 REM DATOS DE SALIDA
1190 REM
1200 REM X    --> SI X>0  X=ELEMENT
O DE LA TABLA DONDE ESTA N$
1210 REM      --> SI X<0  N$ NO EST
A EN LA TABLA
1220 REM
1230 L1=1
1240 X=INT((L1+L2)/2)
1250 IF T$(X)=N$ THEN RETURN
1260 IF L2<L1THEN X=-X:RETURN
1270 IF N$<T$(X) THEN L2=X-1
1280 IF N$>T$(X) THEN L1=X+1
1290 GOTO 1240
1310 REM ---------------------------
------------------------
1320 REM *** CONSTRUCCION DE UNA TA
BLA ORDENADA DE 100 ELEMENTOS ***
1325 DIM T$(100)
1330 FOR I=1 TO 10
1340 FOR J=1 TO 10
1350 T$((I-1)*10+J)=CHR$(I+64)+CHR$
(J+64)
1360 NEXT J
1370 NEXT I
1372 FOR I=1 TO 100
1374 PRINT T$(I)
1376 NEXT
1380 RETURN
1390 REM ---------------------------
------------------------
READY.
```

Líneas 1100-1290: Rutina de búsqueda dicotómica.

Líneas 1320-1380: Creación automática de una tabla de 100 palabras de 2 letras ordenada alfabéticamente.

FUNCIÓN MÓDULO

NOTA DE LA REDACCIÓN DE CLUB COMMODORE: Por un lapsus de esos que aún no nos explicamos, en nuestra edición del mes de Julio se «despistó» una parte del texto del artículo correspondiente de la sección de SOFTWARE DE BASE. Para ser justos, hacemos constar que Ernesto no ha tenido nada que ver con este fallo. Aunque con cierto retraso, publicamos aquí el texto correspondiente junto con la correspondiente rutina. Pedimos montones de disculpas. HABLA ERNESTO:

«Ésta es una función que sólo he encontrado en una versión de BASIC pero que considero de gran utilidad. Se trata de una función que nos permite obtener la parte fraccionaria y el resto de una división. De momento os doy la rutina y en el número próximo (o sea, el pasado Septiembre, para ser claros) veréis una aplicación práctica, ya que aparecerá formando parte de varias funciones.»

```
PROGRAMA:12MOD


9000 REM MOD
9010 REM
9020 REM DATOS DE ENTRADA :
9030 REM
9040 REM X1=DIVIDENDO
9050 REM X2=DIVISOR
9060 REM
9070 REM DATOS DE SALIDA
9080 REM
9090 REM X3=PARTE FRACCIONARIA
9100 REM X4=RESTO
9110 REM
9120 X3=INT(X1/X2)
9130 X4=X1-X3*X2
9140 X3=(X1/X2)-X3
9150 RETURN
READY.
```

■

claves para interpretar los listados de CLUB COMMODORE

Todos los listados que se publican en esta Revista han sido ejecutados en el modelo correspondiente de la gama de ordenadores COMMODORE. Para facilitar la edición de los mismos en la Revista y para mejorar su legibilidad por parte del usuario, se les ha sometido a ciertas modificaciones mediante un programa escrito especialmente para ello. Para los programas destinados a los ordenadores VIC-20 y COMMODORE 64, en los que se usan frecuentemente las posibilidades gráficas del teclado, se han sustituido los símbolos gráficos que aparecen normalmente en los listados por una serie de letras entre corchetes [] que indican la secuencia de teclas que se deben pulsar para obtener el carácter deseado. A continuación se da una tabla para aclarar la interpretación de las indicaciones entre corchetes:

[CRSRD] = Tecla cursor hacia abajo (sin SHIFT)
[CRSRU] = Tecla cursor hacia arriba (con SHIFT)
[CRSRR] = Tecla cursor a la derecha (sin SHIFT)
[CRSRL] = Tecla cursor a la izquierda (con SHIFT)
[HOME] = Tecla CLR/HOME (sin SHIFT)
[CLR] = Tecla CLR/HOME (con SHIFT)

Las indicaciones [BLK] a [YEL] corresponden a la pulsacion de las teclas de 1 a 8 junto a la tecla CTRL. Lo mismo sucede con [RVSON] y [RVSOF] respecto a la tecla CTRL y las teclas 9 y 0.

El resto de las indicaciones constan de la parte COMM o SHIF seguidas de una letra, número o símbolo — por ejemplo [COMM+] o [SHIFA] —. Esto indica que para obtener el gráfico necesario en el programa deben pulsarse simultáneamente las teclas COMMODORE (la que lleva el logotipo) o una de SHIFT y la tecla indicada por la letra, el número o el símbolo, en el ejemplo anterior: COMMODORE y + o SHIFT y A, respectivamente. ■

TRUCOS

"mezcla" de diferentes programas

una solución utilizable tanto en el COMMODORE-64 como en el VIC-20

por P. MASATS

Un problema relacionado con el COMMODORE 64 es el de las dimensiones de los programas que suele manejar y que está preocupando al equipo redactor de nuestra Revista pues corremos el peligro de que los listados de estos programas se nos «coman» las páginas de que disponemos (¡se admiten ideas!), además de consumir el cupo de paciencia del sufrido lector/usuario. Pero no vamos a hablar de los listados del COMMODORE 64 sino de un procedimiento que permite «mezclar» («MERGE» en la jerga inglesa del «rrrollo» informático) partes de diferentes programas en uno solo, para evitar el trabajo de volver a entrar subrutinas de utilidad general que se repiten frecuentemente. Como curiosidad diremos que este procedimiento SIRVE EXACTAMENTE IGUAL PARA VIC-20 Y COMMODORE 64 y además permite el intercambio de programas entre los dos equipos. El procedimiento es como sigue:

(**NOTA.**—La única precaución previa consiste en que la rutina común tenga números de línea diferentes a los del programa con el que se vaya a mezclar. Esto puede conseguirse fácilmente dándole unos números de línea lo suficientemente altos para que queden fuera de los programas que utilizamos normalmente y/o dejando sitio en estos en los lugares en que deban situarse la o las subrutinas «standard».)

Cuando tengamos la rutina que deberemos mezclar con otros programas a escribir en el futuro en la memoria del equipo como un programa en BASIC haremos (todo el procedimiento de mezcla se realiza utilizando el cassette):

OPEN 1,1,1,«NOMBRE DEL PROGRAMA»:CMD1:LIST (RETURN)

«NOMBRE DEL PROGRAMA» es el nombre con el que se archivará la subrutina a mezclar.

Aquí se supone que tenemos una cinta en el aparato de cassette y que seguimos el mismo procedimiento que en la grabación de un programa normal. Cuando se para la cinta y aparece el READY teclee:

PRINT # 1:CLOSE1 (RETURN)

Cuando la cinta vuelva a pararse se puede retirar de la unidad de grabación. Para mezclarla con un programa en memoria:

Colocar la cinta que contiene la subrutina en la unidad de lectura y entrar:

POKE 19,1:OPEN 1 (RETURN)

En el caso del COMMODORE 64 se borrará la pantalla mientras el equipo busca el fichero. Cuando lo encuentre, aparecerá el mensaje FOUND NOMBRE DEL PROGRAMA durante algunos segundos. Si se desea acelerar el proceso hay que pulsar la tecla que lleva el logotipo de COMMODORE. A continuación hay que realizar cuidadosamente las siguientes operaciones (no intente entenderlas. Si le sirve de consuelo le diremos que nosotros no lo hemos conseguido):

Cuando vuelva a aparecer el mensaje READY (y en el caso del 64 resucite la pantalla), limpie la pantalla con las teclas SHIFT y HOME. Pulse **TRES VECES** la tecla de cursor hacia abajo y entre:

PRINTCHR$(19) :POKE198,1:
POKE631,13 :POKE153,1 (RETURN)

La cinta se moverá y finalmente aparecerá en pantalla un surtido de mensajes de error. Prescinda olímpicamente de ellos (¡aprovéchese!, ésta es la única ocasión en la que podrá burlarse impunemente de uno o varios mensajes de error. Sólo por esto, ya vale la pena utilizar este procedimiento). Seguidamente pulse:

CLOSE 1 (RETURN)

Y los dos programas se han mezclado. Para evitar inmerecidas atribuciones de paternidad diremos que este procedimiento se debe a Jim Butterfield que, a su vez, cita como autor a Brad Templeton. A la vista del procedimiento anterior sería interesante ver a Brad «en acción». ■

código máquina del 6502 (II)

por **P. MASATS**



UNA OJEADA AL FÍSICO DEL 6502

Muchos de nuestros lectores deben estar familiarizados con el par de palabras de la jerga informática que parecen creadas con el único fin de confundir al principiante: HARDWARE y SOFTWARE. Estos son dos términos del inglés que designan de manera concisa las dos partes principales en que se divide un ordenador: la parte física — circuitos integrados, placas de circuito impreso, fuentes de alimentación, etc... — que se denomina HARDWARE, y la parte lógica que es de naturaleza algo abstracta, pues consiste en los programas que hacen que el conjunto físico (el HARDWARE) realice las funciones propias de un ordenador.

En esta serie de artículos vamos a tratar del lenguaje más elemental (y el más laborioso) que puede manejar el microprocesador 6502, el que se conoce como CÓDIGO-MÁQUINA. Por lo tanto, vamos a analizar aspectos estrictamente de SOFTWARE... Al menos éste será el cuerpo principal de la información que vamos a suministrar al lector porque, dado que en CÓDIGO-MÁQUINA se manejan de forma casi directa las características del microprocesador, no vemos que sea posible evitar un somero análisis de la estructura física del 6502.

EL «CHIP» 6502

Su aspecto esquemático es el de la figura 1, donde se puede ver que posee 40 patillas o conexiones numeradas del 1 al 40 en sentido inverso al de las agujas del reloj. La comunicación entre este microprocesador y sus periféricos se realiza a través de estas conexiones, entendiendo por periféricos, de ahora en adelante, no las unidades de disco, cassette, impresora, etc..., sino los otros circuitos integrados (memorias, generador de video, unidades de entrada/salida, etc...) que componen el ordenador en sí mismo. En este sentido — una vez analizado en detalle el conjunto de bloques que componen el ordenador básico — se discutirá la estructura de estos periféricos.

Para realizar la adecuada comunicación microprocesador-periféricos se utiliza un truco que consiste en intercambiar la información con un código especial que incluye un tipo de numeración específico. Se trata de que las señales que deben circular por la totalidad de los circuitos que componen el ordenador estén exentas de lo que en electrónica se conoce como «ruido» (noise) que puede tener orígenes diversos (inducción desde otros dispositivos, por ejemplo) y consecuencias siempre funestas para el buen funcionamiento del equipo. En Radio, por ejemplo, el ruido es esa señal que, superpuesta a la de una emisora lejana, dificulta o llega a impedir que entendamos lo que el locutor está diciendo. Con el fin de luchar contra el ruido en el caso del 6502 (y de muchos otros dispositivos digitales) las tensiones que se pueden encontrar en sus conexiones pueden tener sólo dos valores «válidos»: 0 y 5 voltios que corresponden, respectivamente, a los dígitos 0 y 1 de la notación en base 2 o binaria (más adelante revisaremos en detalle esta base de numeración). De un dispositivo que trabaja con este tipo de tensiones se dice que pertenece a la familia lógica TTL (de Transistor-Transistor-Logic, o lógica transistor-transistor) aunque más exacto resulta decir que sus conexiones son compatibles TTL. Así pues en todas las «patillas» de nuestro protagonista en un determinado momento encontraremos o un 0 (una tensión de 0 V) o un 1 (una tensión de 5 V). Es necesario hacer una pequeña salvedad: se trata de que estas tensiones no son exactas y de que tanto el 0 como el 1 lógicos tienen un cierto margen dentro del cual serán aceptados como tales: para el 0 de 0 a 0,8 V y para el 1 de 2,4 V en adelante.

PARADA PARA TOMAR EL «BUS»

Detenemos un momento la «marcha» de nuestras explicaciones para analizar un concepto fundamental de la estructura de todo microprocesador: el **bus.**

En un sistema controlado por un microprocesador, éste debe comunicarse con sus periféricos. Por ejemplo, supongamos que estamos utilizando el intérprete BASIC y queremos hacer POKE 36879,8 o sea que queremos escribir el valor ocho en la posición de memoria 36879, luego el microprocesador debe poder decirle a un chip determinando **«dónde»** (en la posición

(continúa en la pág. 12)

El mejor ordenador personal del Mundo



El COMMODORE 64

Este es el nuevo ordenador personal COMMODORE 64. Un gigante de 40 cm, con un precio casi tan pequeño como su tamaño.

Nadie hasta ahora había logrado ofrecerle 64 K de memoria, 40 columnas en pantalla; 8 sprites y un sonido de auténtica maravilla por sólo 110.000,— ptas. Claro que tampoco todo el mundo es el líder mundial en microordenadores.

COMMODORE sabe perfectamente que para seguir siendo el número uno, tiene que estar constantemente en vanguardia.
De calidad. De precios. De todo. Para ello investigamos constantemente.

Afortunadamente nuestra labor se ve plenamente recompensada cuando vemos, como lo demuestra el cuadro comparativo, que nuestro más directo competidor cuesta nada menos que un 100% más caro. Y ello sin reunir todos los adelantos técnicos del COMMODORE 64.

---

1. Capacidad total de memoria RAM de 64 K. Interpretador BASIC extendido y sistema operativo residentes en ROM.

2. Dotado del más potente chip sintetizador de sonido diseñado hasta hoy, el COMMODORE 64 ofrece 3 voces totalmente independientes con una gama de 9 octavas. El programa puede controlar la envolvente, la afinación y la forma de onda de cada voz,

convirtiendo al COMMODORE 64 en el mejor simulador de instrumentos.

3. Conectable directamente a toda una gama de periféricos, incluyendo unidad de discos, impresora de matriz de puntos o de margarita, plotter, comunicaciones locales y remotas..., y mucho más.

4. Pantalla de alta resolución en color con 320 × 200 puntos directamente direccionables. Capacidad en modo carácter de 25 líneas por 40 columnas.

5. El chip de video, único en su género, permite el uso de 8 «Sprites» (figuras móviles en alta resolución y color). Los «Sprites» pueden moverse independientemente por programa de «pixel» en «pixel».

6. A cada «Sprite» se le asigna por programa un nivel de prioridad en caso de cruce con otro, consiguiendo efectos tridimensionales, existiendo también detección automática de colisiones.

7. Teclado profesional con mayúsculas y minúsculas, más 62 caracteres gráficos, todos ellos disponibles en el teclado y visualizables en 16 colores, en forma normal o bien en video invertido.

8. Encontrará a su disposición una completa gama de programas profesionales, incluyendo proceso de textos, sistemas de información, modelos financieros, contabilidad y muchas más aplicaciones.

9. Están en fase de desarrollo asimismo otros lenguajes tales como LOGO, UCSD PASCAL, COMAL, ASSEMBLER, etc. Todos los programas existentes de la gama COMMODORE, desde el VIC-20 hasta los modelos CBM pueden ser adaptados fácilmente.

10. Posibilidad de inserción de cartuchos con programas grabados en ROM, tanto profesionales como para educación y ocio.

11. Opción de un segundo procesador Z-80 para trabajar con sistema operativo CP/M (R).

EL COMMODORE 64 Y SU MAS DIRECTO COMPETIDOR

OPCION DE BASE	COMMODORE 64	Más directo competidor
Precio	110.000,— ptas.	El doble
Memoria usuario	64 K	48 K
Teclado profesional	SI	SI
Teclado con caracteres gráficos	SI	NO
Minúsculas	SI	NO
Teclas de función	SI	NO
Máxima capacidad disco	170 K a 1 M	143 K
AUDIO		
Generador de sonido	SI	SI
Sintetizador de música	SI	NO
Salida HI-FI	SI	NO
VIDEO		
Salida monitor	SI	SI
Salida para TV	SI	EXTRA
PERIFERICOS		
Cassette	SI	SI
Periféricos inteligentes	SI	SI
Bus serie	SI	NO
SOFTWARE		
Opción CP/M (R)	SI	SI
Ranura cartucho externo	SI	NO

commodore
COMPUTER

PARA MAS INFORMACION
DEL COMMODORE 64,
LLAMAR O ESCRIBIR A:
MICROELECTRONICA Y CONTROL
c/ Taquígrafo Serra, 7, 5º. Barcelona-29
Tel. (93) 250 51 03
c/ Princesa, 47, 3º, G. Madrid-8
Tel. (91) 248 95 70

Nombre ....
Dirección ....
Tel. ....
Población ....

SLOGAN

código máquina del 6502

(viene de la pág. 9)

de memoria 36879) y **«qué»** valor debe almacenar. Por lo tanto, a través de todo el sistema debe «circular» un medio de comunicación que lleve estos datos. Esta función la cumplen los «buses» del sistema los cuales, bajo control del microprocesador sirven para canalizar y conducir estas informaciones. Su forma física es la de un conjunto de conductores, cada uno de los cuales lleva información de un solo bit que, como el lector debe saber, es la unidad de información del sistema binario o de base dos, por lo cual dicho bit puede tener sólo dos valores: el 0 ó el 1.

El «bus» que le dice al periférico «dónde» debe almacenar un valor — o de «dónde» debe leerse una cantidad previamente almacenada — se llama **«bus de direcciones»** y en el caso del 6502 tiene 16 bits, que si calculamos el número de combinaciones diferentes de ceros y unos que pueden presentar nos dan 2 ↑16 = 65536 que es la máxima cantidad de posiciones de memoria que este microprocesador puede manejar.

	Pin	Pin	
Vss	1	40	← $\overline{RES}$
RDY →	2	39	→ Ø2 (OUT)
Ø1 (OUT) ←	3	38	← S.O.
$\overline{IRQ}$ →	4	37	← Ø0 (IN)
N.C.	5	36	N.C.
$\overline{NMI}$ →	6	35	N.C.
SYNC ←	7	34	→ R/W
Vcc	8	33	DBØ ↔
ABØ ←	9	32	DB1 ↔
AB1 ←	10	31	DB2 ↔
AB2 ←	11	30	DB3 ↔
AB3 ←	12	29	DB4 ↔
AB4 ←	13	28	DB5 ↔
AB5 ←	14	27	DB6 ↔
AB6 ←	15	26	DB7 ↔
AB7 ←	16	25	AB15 →
AB8 ←	17	24	AB14 →
AB9 ←	18	23	AB13 →
AB10 ←	19	22	AB12 →
AB11 ←	20	21	Vss

MCS6502

fig. 1 - NC = NO CONECTADO

Para el caso del «qué», existe un «bus de datos» que realiza la misma función. El microprocesador pone en él el valor que la memoria debe almacenar en la posición expresada por el «bus de direcciones». En el caso del 6502, este «bus» tiene ocho conductores por lo que el número de combinaciones diferentes que pueden tomar estos ocho conductores es de 2 ↑8, o sea 256. Se dice que éste es un «bus de ocho bits», lo que determina una de las principales clasificaciones de los microprocesadores y, por lo tanto, conocemos al 6502 como **un microprocesador de ocho bits.** Existen, para acabar con este tema, algunas consideraciones complementarias, a saber: no sólo es necesario establecer comunicación en torno al «dónde» y al «qué», sino que es necesario establecer «cuándo» la información presente en los «buses» es válida y «cómo» hay que tratarla. De esto se encarga un tercer «bus» denominado «bus de control». Otro asunto importante es el de aclarar cómo se establece el control de los valores que se transportan en los «buses», pues mientras el «bus de direcciones» está bajo control exclusivo del microprocesador (existe una excepción que veremos más adelante), en el «bus de datos» la memoria debe depositar el valor que tiene almacenado (por ejemplo, en una operación de PEEK), por lo que este «bus» se dice que es **bidireccional.** En el caso del «bus de control» hay un poco de todo, dado que más que un «bus» propiamente dicho es un conjunto de señales casi independientes.

En torno al término «bus» creemos necesario aclarar un detalle en cuanto a la terminología: la palabra «bus» parece sugerir un medio de transporte de información a lo largo del cual se mueven las señales, cuando, por propagarse éstas a la velocidad de la luz, dicho movimiento es instantáneo. En este sentido, recomendamos al lector que imagine el «bus» como un depósito donde las diferentes partes del sistema depositan información mientras otras la toman. De todas maneras la denominación de «bus» es universal y nadie — que sepamos — ha propuesto cambiarla.

(Continuará)

MULTICENTRO DE INFORMATICA

Chips & Tips

PUERTO RICO, 21-23 - MADRID-16 TEL. 250 74 02 - 250 74 04

commodore VIC-20

JUEGOS

- COMECOCOS. 3,5K. A/R. G/E. JY. EXTRAORDINARIA VERSION DEL POPULAR PUCKMAN. COLOR Y MOVILIDAD INSUPERABLE . 1.900
- VICGAMON. +3K. JUEGO DE INTELIGENCIA QUE LE MANTENDRA EN TENSION HASTA DERROTAR A SU VIC 1.800
- ASTEROIDS WAR. 3,5K. C/M. A/R. G/E. JY. ESPECTACULAR BATALLA GALACTICA CONTRA LA NUBE PROTONICA EN 3 DIMENSIONES 1.800
- FROGGER. +3K y 3,5K. C/M. A/R. G/E. JY. ULTIMA NOVEDAD EN EE.UU. CRUZAR EL PELIGROSO RIO Y LA AUTOPISTA SUICIDA 2.000
- RATMAN. +8K. C/M. A/R. G/E. DE LA BOVEDA CELESTE DESCENDERAN EXTRAÑAS RATAS ATOMICAS. ESPECTACULAR ANIMACION 1.900
- SHARK ATTACK. 3,5K. C/M. A/R. JY. EN MEDIO DEL OCEANO SERA ATACADO POR LOS PELIGROSOS TIBURONES. DEFIENDASE CON SU RED ATOMICA.................................. 1.900
- ROX III. 3,5K Y+8K. C/M.A/R. G/E. JY. DESDE SU SOFISTICADA BASE LUNAR DEFIENDA SU PLANETA DEL ATAQUE DE LOS UFOS 1.800
- .ULTISOUND SYNTHETIZER. 3,5K. ¿UN ORGANO EN SU VIC? ¿CON ACOMPAÑAMIENTO, BATERIA Y EFECTOS ESPECIALES? . . 1.900
- SKI-RUN. 3,5K. C/M. A/R. G/E. DESLICESE POR LAS HELADAS PISTAS DE COMPETICION. SLALOM, S/GIGANTE, DESCENSO. 9 NIVELES 1.800
- FIREBIRD. (SPACE PHREEKS). 3,5K. C/M. A/R. G/E. JY. AÑO 3.010. VD. ES EL UNICO SUPERVIVIENTE DE LA BATALLA DE RIGELLIAN. DEBERA COLONIZAR OTRO PLANETA Y LUCHAR CONTRA LAS CRIATURAS GALACTICAS..................... 1.900
- BREAKOUT. 3,5K. CONSIGA DESTRUIR LA PARED DE LADRILLOS MULTICOLORES CON LA BOLA MAGICA. INCLUYE "MASTERMIND". 1.600
- AJEDREZ. PRIMERA VERSION EN CASSETTE CON GRAFICOS EN ALTA RESOLUCION. BASTANTES NIVELES DE JUEGO. (STANDARD) . 2.800
- SHADOWFAX. INCREIBLES GRAFICOS ANIMADOS. EL CABALLERO DE LAS SOMBRAS EN LUCHA CONTRA LOS JINETES DEL TIRANO INVASOR. (STANDARD)................................ 1.900
- SNAKE. COLORIDO, MOVIMIENTOS Y GRAFICOS EXCEPCIONALES. VERSION DEL FAMOSO JUEGO DE LAS SERPIENTES (SNAKE). (STANDARD).. 1.900
- MYRIAD. +3K. C/M. A/R. G/E. JY. LA MAS ESPECTACULAR AERONAVE PARA DESTRUIR LAS CRIATURAS COSMICAS EN SU VIAJE GALACTICO.. 2.000
- COSMIADS. 3,5K. C/M. A/R. G/E. JY. VERSION ULTRARRAPIDA DEL MUNDIALMENTE FAMOSO JUEGO "GALAXIANS". INCREIBLES EFECTOS SONOROS.. 1.700
- BLITZRIEG. 3,5K. C/M. A/R. G/E. JY. DESTRUYA LA CIUDAD ENEMIGA CON SU BOMBARDERO. 25 NIVELES DE JUEGO 1.600
- DEFENSA. +8K. C/M. A/R. G/E. JY. N.º 1 EN INGLATERRA, COMO GUERRERO GALACTICO DEBE DEFENDER A LOS HUMANOIDES CONTRA LOS ENEMIGOS CIBERNETICOS. 9 NIVELES DE JUEGO. ESPECTACULAR NAVE Y SONIDOS 2.000
- VIC PANIC. 3,5K. C/M. A/R. G/E. JY. VERSION DEL POPULAR "SPACE PANIC". ESCALE LAS LADERAS Y HUYA DE LOS MONSTRUOS 1.800
- SKRAMBLE. 3,5K. C/M. A/R. G/E. JY. ATRAVESANDO LOS TEMIBLES PASADIZOS INTERESTELARES DESTRUYA LAS BASES ENEMIGAS 1.900
- 3D LABYRINTH. +8K. C/M. A/R. EXTRAORDINARIO LABERINTO TRIDIMENSIONAL. ¿SERA CAPAZ DE SALIR DE EL? UNO O VARIOS JUGADORES 1.800
- GOLF. 3,5K. RECORRIDO DE 9 HOYOS PERO ATENCION A LOS OBSTACULOS: ARBOLES, LAGOS, ETC. INCLUYE VIC MUSIC Y PIANO 1.600
- CARRERA DE BUGGYS. 3,5K. C/M. A/R. G/E. ESPECTACULAR RECORRIDO. ACELERADOR. DECELERACION. 9 NIVELES 1.800
- GRIDRUNNER. 3,5K. C/M. A/R. G/E. JY. IMPRESIONANTE VERSION LLENA DE COLORIDO, MOVILIDAD Y SONIDO DEL POPULAR "CENTIPEDE".. 1.900
- HI-RES. 3,5K. GRAN JUEGO DEMO/UTILIDAD PARA REALIZAR EN PANTALLA GRAFICOS EN ALTA RESOLUCION. INCLUYE GEN. CARACTERES... 1.500
- ABDUCTOR. LAS CRIATURAS COSMICAS DEL PLANETA "ALPHA I" INTENTARAN SECUESTRAR A LOS HUMANOIDES PARA CONSEGUIR ENERGIA E INTELIGENCIA SUPERIORES. TU MISION SERA DEFENDER TU PLANETA Y DESTRUIR LAS NAVES ABDUCTORAS. (STANDARD). 1.800
- TRAXX. VERSION DEL CONOCIDO JUEGO "AMIDAR"; MEZCLA DEL POPULAR "PACKMAN" Y DEL JUEGO "QUIX". 100% CODIGO MAQUINA. GRAFICOS EN ALTA RESOLUCION. ESPECTACULAR SONIDO Y COLOR. 8K DE MEMORIA 2.000

UTILIDADES

- VIC PRINT. +8K. EXTRAORDINARIO Y SENCILLO PROCESADOR DE TEXTOS. TABULACION, MAQUETACION, CABECERAS, COPIAS. CASS O DISK... 2.000
- VIC LABEL. +8K. EN COMBINACION CON VIC PRINT, ELABORA ETIQUETAS PARA DIRECCIONES........................... 1.900
- VIC POST. +8K. ELABORA LETRAS Y TEXTOS ESPECIALES EN TAMAÑO Y FORMA PARA POSTERS, LISTAS DE PRECIOS, ETC. . . 2.900
- VIC CALC. HERRAMIENTA DE CALCULO QUE SUSTITUYE AL LAPIZ, PAPEL Y CALCULADORA, REALIZA COMPLEJOS MODELOS FINANCIEROS CON POSIBILIDAD DE AJUSTARLO A OTROS PARAMETROS CON SOLO PULSAR UNA TECLA. 16K DE MEMORIA............ 3.200
- VIC BASE. 16K. POTENTE BANCO DE DATOS. 255 CARACTERES, MAS DE 25 CAMPOS. CAMBIO Y LOCALIZACION, SALIDA IMPRESORA... 3.200
  OPCION 3,5K.. 1.800
- GRAPHVICS. +3K. AÑADE 18 POTENTES COMANDOS PARA POSICIONAR PUNTOS, DIBUJAR LINEAS Y TEXTOS EN ALTA RESOLUCION (152×160) 2.200
- GRAPH EDITOR & SOFTKEY 24. 3,5K. AMBOS PROGRAMAS PERMITEN DISEÑAR HASTA 64 CARACTERES PARA INCORPORARLOS A SUS PROPIOS PROGRAMAS Y JUEGOS 2.000

EDUCATIVOS

- QUIZ-MASTER. +3K. EL MAS ESPECTACULAR AVANCE EDUCATIVO. PERMITE LA CORRECCION Y PUNTUACION DE TODAS LAS RESPUESTAS QUE RECIBE EL ORDENADOR.......................
- QUIZ SET-UP. EN TANDEM CON QUIZ—MASTER PERMITE LA ELABORACION POR EL USUARIO DE TODO TIPO DE PREGUNTAS Y CUESTIONES EDUCATIVAS O DE ENTRETENIMIENTO, EGB, IDIOMAS, MATEMATICAS, HISTORIA, GEOGRAFIA, ETC. CREANDO UN AGIL Y ATRACTIVO SISTEMA DOMESTICO/EDUCATIVO ..... } 3.200
- FACEMAKER. 16K. CARICATURANDO EL ROSTRO DE SUS COMPAÑEROS Y AMIGOS EL VIC 20 PONDRA A PRUEBA EL VOCABULARIO Y LA ATENCION DEL NIÑO................................ 2.000
- NUMBER CHASER. 16K. PROGRAMA PARA PRACTICAS DE MULTIPLICACION CON CARRERAS DE COCHES, ADELANTA, FRENA, ACELERA SEGUN LAS RESPUESTAS. 4 NIVELES DE DIFICULTAD . . . 2.000
- NUMBER GULPER. 16K. JUEGO EDUCACIONAL DE COMPETICION CON NUMEROS PARA SUMA, RESTA, MULTIPLICACION Y DIVISION 2.000
- WE WANT TO COUNT. 16K PROGRAMA PARA NIÑOS A PARTIR DE TRES AÑOS, INVASORES, CARRERAS, ETC. 2.000
- TWISTER. 16K. JUEGO DE LOGICA Y CONCENTRACION. PUZZLES GEOMETRICOS CON SONIDO Y COLOR.......................... 2.000

LIBROS

VIC REVEALED....................	2.200	ASSEMBLER	2.000	ZAP! POW! BOOM!	1.800
GETTIN ACQUAINTED WITH YOUR VIC 20	1.800	SYNPHONY MELANCHOLY COMP.....	1.800	VIC INNOVATIVE..................	2.000
50 PROGRAMAS LISTADOS I.........	1.500	50 PROGRAMAS LISTADOS II	1.500	50 PROGRAMAS LISTADOS III........	1.500

FICHEROS CBM (III)



estructura del disco

por **MANUEL AMADO**
(M.E.C.-SOFT.)

Antes de pasar a ver las técnicas de acceso directo, que nos van a servir para examinar a fondo la organización de los datos almacenados en un disquete o para realizar un fichero de acceso directo o RANDOM hay que conocer cuál es la estructura de los datos dentro del disco. En este caso, el guión para los dos próximos artículos es el siguiente:

1. Introducción a la estructura del disco.
2. Técnicas de acceso directo y ficheros RANDOM.
3. Distribución de los ficheros en disco (puesto que ya sabemos acceder directamente a un determinado lugar del disco, vamos a estudiar profundamente cómo se organiza el fichero físicamente en el disco, cómo se distribuyen sus datos, la forma de reseguir un fichero y recuperación de datos de ficheros dañados).

1. INTRODUCCIÓN A LA ESTRUCTURA DEL DISCO

Los disquetes, una vez formateados, están divididos en diferentes pistas, 35 en el caso de los modelos CBM 3040, 4040, 1540 y 1541, 77 en el modelo CBM 8050 y 144 en el modelo CBM 8250. A su vez, cada pista está dividida en sectores con 256 bytes de datos cada uno (ver figura 1). El número de sectores por pista depende del número de la pista, puesto que en las pistas más exteriores al centro del disquete de número más bajo, como disponen de más superficie física que las pistas centrales, contendrán más sectores que estas últimas. En la tabla 1 puede verse la distribución de sectores o bloques por pistas para diferentes modelos de discos CBM.

Por otro lado, el **DOS,** sistema operativo de disco (DISK OPERATING SYSTEM), se encarga de distribuir los datos de los diferentes ficheros en el disquete y gestionar e interpretar todas las órdenes que le son enviadas de la CPU.

Cuando creamos un fichero en el disquete mediante la orden OPEN o DOPEN adecuada, y posteriormente escribimos datos en él, o bien grabamos un programa en el disco, etc., el **DOS** ha de saber en qué lugar del disquete va a guardar el programa, los datos del fichero, etc. Pues bien, para ello se reserva para su uso propio unos bloques de control, que forman el **DIRECTORIO** y el **BAM.** Veamos qué significado y misión tienen cada uno de ellos:

DIRECTORIO:

Indica el nombre del disquete, su identificador, qué ficheros contiene, y el total de bloques libres que hay en ese disquete. Concretamente, para cada fichero nos da la siguiente información cuando lo visualizamos en pantalla:

— Número de bloques que ocupa el fichero.

— Nombre del fichero.

— Tipo del fichero (Borrado (DEL), Secuencial (SEQ), Programa (PRG), Usuario (USR) o Relativo (REL)).

Además, para cada fichero hay un puntero que indica dónde están los bloques correspondientes, con lo que el DOS puede acceder en cualquier momento a los datos de cualquier fichero.

El directorio se puede visualizar en pantalla mediante el comando DIRECTORY o CATALOG del BASIC 4 o bien **ejecutando un LOAD "$0",8 en el caso de estar trabajando con BASIC 2.**

El **DOS** reserva 29 bloques para DIRECTORIO en el caso del 8050, con lo cual podemos controlar y tener en un disquete hasta 224 ficheros. En un próximo artículo detallaré más a fondo la estructura byte a byte del directorio, **...pero antes ¡hay que aprender** a leer estos bytes!

EL **BAM:**

Son las siglas en inglés de MAPA DE BLOQUES DISPONIBLES (BLOCK AVAILABILITY MAP), y por él, el **DOS** sabe si un determinado bloque del disco está libre u ocupado, o sea, si forma o no parte de un fichero. En el caso del 8050, para controlar los 2052 bloques libres, son suficientes dos bloques de **BAM.**

Como habréis podido observar, estamos ante una información binaria, libre u ocupado, sí o no, ... 0 ó 1. Y para guardar esta información, sólo necesitamos UN BIT por bloque o sector del disco. El **BAM,** en sus bloques de datos, informa al **DOS** de si un bloque o sector determinado está libre si el BIT correspondiente a dicho fichero en el **BAM** está a 1. Si está a 0 significará que está ya asignado a un fichero y que no puede ser usado por otro.

2. TÉCNICAS DE ACCESO DIRECTO

A) INTRODUCCIÓN Y FILOSOFÍA DEL **DOS** COMMODORE:

Antes de empezar a describir los comandos, órdenes, etc., etc. y más etc., para realizar el acceso directo a un determinado «track» y sector, veamos cómo se realiza la comunicación entre la CPU y los datos que hay en el disquete.

(sigue el texto en la pág. 16)

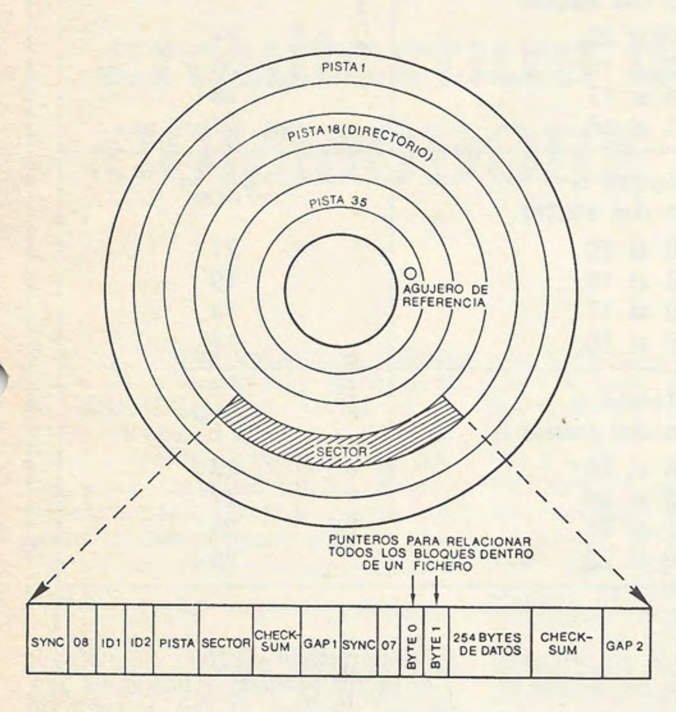


Fig. 1A - Formato de los discos grabados en unidades 2040, 3040, 4040, 1540 y 1541.

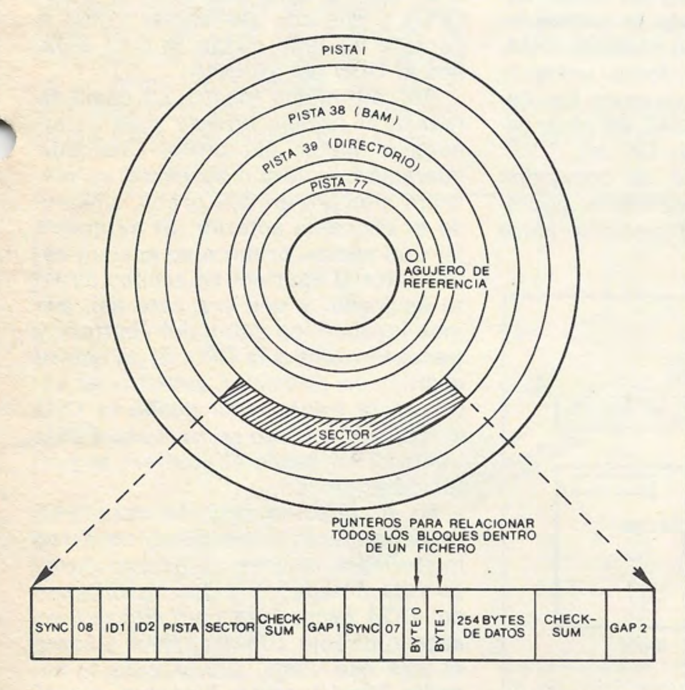


Fig. 1B - Formato de los discos grabados en unidades 8050 y 8250 (dos caras, la segunda sin BAM ni DIRECTORIO).

TABLA 1 - **DISTRIBUCIÓN DE BLOQUES POR PISTAS**

2040, 3040 Número de pista	Bloque o Rango del sector	Total
1 al 17	0 al 20	21
18 al 24	0 al 19	20
25 al 30	0 al 17	18
31 al 35	0 al 16	17
4040,1540, 1541 Número de pista	**Bloque o Rango del sector**	**Total**
1 al 17	0 al 20	21
18 al 24	0 al 18	19
25 al 30	0 al 17	18
31 al 35	0 al 16	17
8050 Número de pista	**Bloque o Rango del sector**	**Total**
1 al 39	0 al 28	29
40 al 53	0 al 26	27
54 al 64	0 al 24	25
65 al 77	0 al 22	23

— Comunicación entre CPU y «floppy»:

La comunicación entre la CPU y cualquier periférico CBM a través del BUS IEEE se establece abriendo un fichero o CANAL DE COMUNICACIÓN mediante la orden OPEN del BASIC.

Observemos la figura 2. En ella vemos tres partes FÍSICAS diferentes que intervienen en el ya famoso proceso de comunicación:

— El disquete, soporte físico magnético, en donde ESTÁ el bloque que queremos tratar.

— Un buffer de RAM. Desde este buffer se cargarán y grabarán los bloques del disquete.

— La CPU, en donde se procesan los datos provenientes del bloque accedido.

El acceso al bloque del disquete, siguiendo el anterior esquema físico, se realiza del siguiente modo:

Abriendo un canal especial de acceso directo, mediante un OPEN especial, tenemos abierta la comunicación entre la CPU y el BUFFER RAM. Para leer y escribir datos entre la CPU y el BUFFER, usaremos los comandos normales BASIC en cada caso (PRINT #, INPUT #, GET #).

Mediante una serie de comandos del **DOS,** que luego detallaré, podremos intercambiar información entre un determinado «buffer» del «floppy» y el sector deseado, y movernos dentro del «buffer». O sea, el traspaso de información entre «buffer» y un bloque del disquete se realiza a través de comandos **DOS,** que se envían **mediante un canal de órdenes especial,** que se abre mediante la orden OPEN y que con sentencias PRINT # permite enviarle desde la CPU **órdenes** al **DOS** del «floppy».

Así, a grandes rasgos, es cómo se realiza el acceso directo a un determinado sector del disco. Cualquier operación, lectura o escritura, se realizará siempre en dos pasos, siguiendo el esquema anterior. Si se quiere leer un sector, primero se enviará este sector al «buffer» de acceso directo asignado, y una vez esté allí, iremos leyendo los datos del «buffer» y traspasándolos a la CPU. Si se quiere escribir en un sector, primero se escribirá la información desde la CPU al «buffer», y luego se traspasará esta información desde el «buffer» al sector determinado.

En el próximo artículo describiré detalladamente cómo se abren estos misteriosos canales de acceso directo y de órdenes, y todas las órdenes del **DOS** necesarias para trabajar en acceso directo. Sin nada más, y hasta el mes que viene, que ustedes lo accedan bien! (perdón, filosofen, que el acceso viene el mes que viene). ■

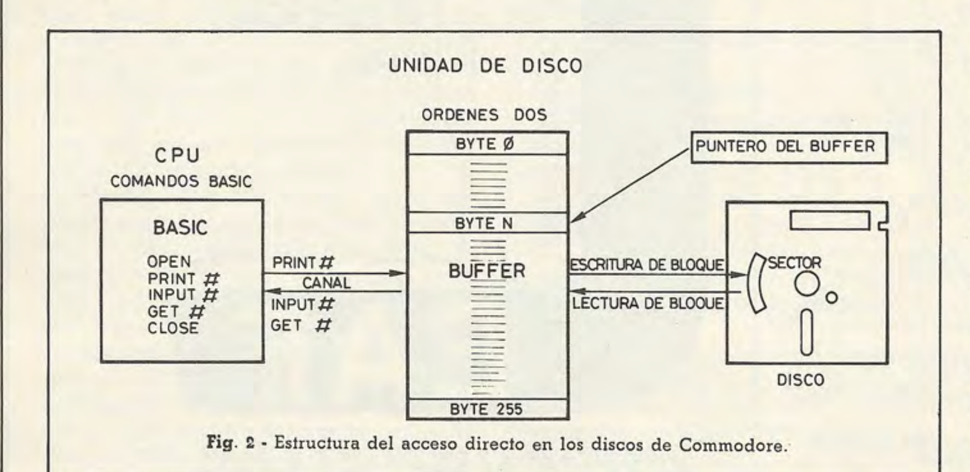


Fig. 2 - Estructura del acceso directo en los discos de Commodore.

COMMODORE 64

arquitectura del COMMODORE 64



por **R. PARDO**

En los últimos tiempos se nos ha hecho una pregunta muy típica: si el COMMODORE 64 tiene 64 Kbytes de RAM: ¿dónde se sitúan las ROM de sistema operativo, intérprete de BASIC y los chips de Entrada/Salida? En este artículo vamos a intentar responder a esta pregunta y dar la imagen correcta de lo que hay dentro de este ordenador.

El COMMODORE 64 dispone de un microprocesador de la familia 65XX, el 6510. Este microprocesador es compatible en software y hardware con el 6502. La única diferencia entre ambos chips consiste en las posiciones $0000 y $0001 del procesador que contienen los registros de control y de dirección de un port de Entrada/Salida integrada en el mismo chip. Para entender cuál es la misión de este port de Entrada/Salida, explicaré el mapa de memoria del 64. El COMMODORE 64 tiene 64 K de memoria RAM que cubren todo el espacio direccionable del microprocesador (16 líneas de dirección). En ciertas zonas de la memoria hay superpuestas secciones de memoria ROM y/o E/S. Al poner en marcha el COMMODORE 64 verá parte de estos 64 K de memoria y las secciones de ROM de BASIC, ROM de KERNAL y E/S (controlador de video [VIC II], controlador de sonido [SID] y los controladores de periféricos). Para dar una idea de la estructura del mapa de memoria del COMMODORE 64, adjunto una representación esquemática de éste (fig. 1).

Las posiciones en las que están estas secciones superpuestas de memoria son:

SECCIÓN	POSICIONES QUE OCUPA
ROM BASIC	$A000-$BFFF
E/S	$D000-$DFFF
ROM CARACTERES	$D000-$DFFF
ROM KERNAL	$E000-$FFFF

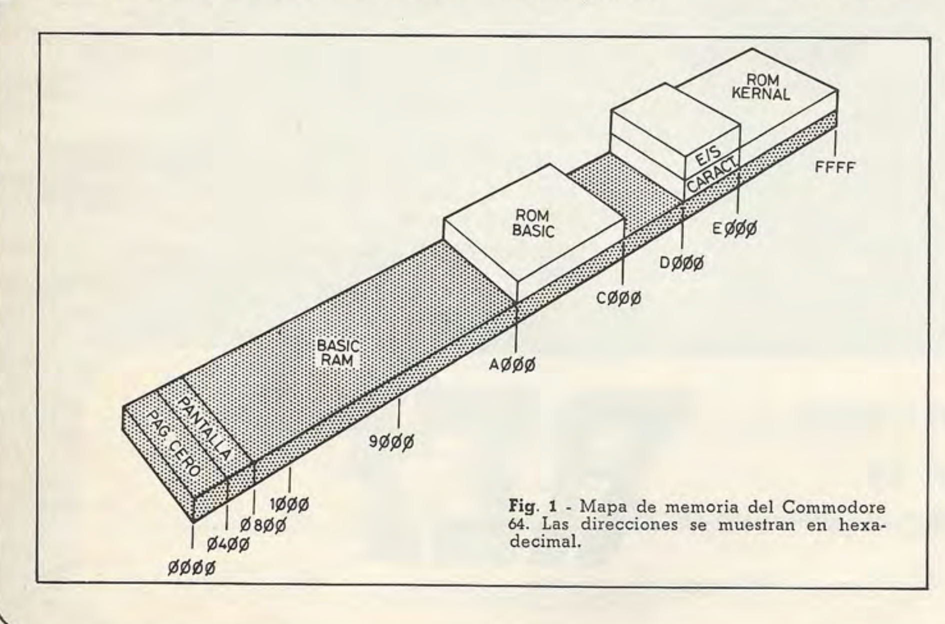


Fig. 1 - Mapa de memoria del Commodore 64. Las direcciones se muestran en hexadecimal.

IMPORTANTE PAPEL DEL PORT. DE ENTRADA/SALIDA

Luego, ¿qué tiene que ver el port de E/S del microprocesador en todo esto y cómo sabe el ordenador qué sección de memoria debe apuntar? Las dos preguntas tienen la misma respuesta, ya que el port de Entrada/Salida que lleva integrado el microprocesador juega un papel importantísimo en la configuración del mapa de memoria del ordenador. Si nosotros jugamos con los tres primeros bits de la posición $0001 advertirá que pasan una serie de cosas raras: el ordenador lo más probable quedará en un estado catatónico que le obligará a desconectar y volver a conectar su máquina. Estos tres primeros bits nos controlan la configuración del mapa de memoria de la siguiente forma:

— El bit 0 (LORAM) controla el espacio de memoria comprendido entre $A000 y $BFFF (ROM de BASIC) y su valor normal es 1. Cuando es 0, el microprocesador verá lo que haya inmediatamente debajo (en este caso 8 K de RAM).

— El bit 1 (HIRAM) controla el espacio de memoria comprendido entre $E000 y $FFFF (ROM de KERNAL) y su valor normal es 1. Cuando sea 0 el microprocesador verá la zona de RAM comprendida entre $E000 y $FFFF.

— El bit 2 (CHAREN) controla el espacio de memoria comprendido entre $D000 y $DFFF (zona de E/S) y su valor normal es 1. Cuando sea 0 el microprocesador verá la ROM de CARACTERES. ¡¡OJO!! Para ver

(continúa en la pág. 19)

Ahora el VIC-20 y CBM 64 pueden comunicarse con Periféricos Commodore

USUARIOS DEL VIC-20 y CBM 64
¿Le gustaría tener acceso a cualquiera de los siguientes perifericos desde su computador?

* Discos de 1/3 megabyte (Commodore 4040)
* Discos de 1 megabyte (Commodore 8050)
* Discos de 2 megabyte (Commodore 8250)
* Discos de 10 megabyte (Commodore 9090 discos duros)
* Impresoras con IEEE y RS 232 matricial y margarita
* Instrumentos IEEE, como voltimetros, plotters, etc.

Ahora ya no se queda Vd. limitado por el VIC y la serie de los 64. Simplemente añadiendo un INTERPOD puede Vd. aumentar ampliamente la potencia de su VIC-20 y usándolo con el nuevo CBM 64, el INTERPOD convertira su computador en un sistema realmente potente.

Con el INTERPOD, el VIC-20 y el CBM 64 son capaces de llevar a cabo un software de calidad y profesional, tales como proceso de datos, Contabilidad, Control de stok y mucho más ...
INTERPOD está capacitado para trabajar con cualquier software. No se necesitan ningún comando extra y no afecta bajo ningún aspecto para nada a su computador.
Usar el INTERPOD es tan simple y fácil como:
* Enchufar el INTERPOD en la salida de serie de su computador, pongalo en funcionamiento y ya está Vd. listo para comunicarse con cualquier periferico de la serie IEEE y cualquier impresora RS232.
ESTO ES EL INTERPOD.

Importador para España:

C/. BALMES. 13
Tel. (971) 24 54 04
Palma



Es un producto de Oxford Computer Systems (Software) Ltd. U.K.

arquitectura del COMMODORE 64

(viene de la pág. 17)

la RAM que está debajo se tendrá que intervenir con interrupciones desde código máquina.

La Tabla 1 muestra todas las combinaciones posibles de memoria y los valores POKE necesarios para las distintas configuraciones.

La última configuración es la que adopta el ordenador automáticamente cuando se conecta a la red. Como podemos ver, EXISTEN estos 64 K de RAM y podemos empezar a imaginar cuántos mapas de memoria o cuántas configuraciones distintas puede adoptar el COMMODORE 64. De todas maneras, no quiero haceros sufrir. El COMMODORE 64 dispone de ¡¡8 MAPAS DE MEMORIA DISTINTOS!!, lo cual nos permite una flexibilidad increíble, ya que las secciones ROM de BASIC y de KERNAL pueden ser sustituidas por ROMs exteriores (en forma de cartucho) o por un lenguaje que se cargue desde disco (como el sistema operativo CP/M), y todo esto sin perder memoria RAM.

(continuará)

TABLA 1

BIT 76543210	POKE 1, X	A000-BFFF BASIC	D000-DFFF E/S	E000-FFFF KERNAL
00110000	48	RAM	RAM	RAM
00110001	49	RAM	CARÁCTER	RAM
00110010	50	RAM	CARÁCTER	KERNAL
00110011	51	BASIC	CARÁCTER	KERNAL
00110100	52	RAM	RAM	RAM
00110101	53	RAM	E/S	RAM
00110110	54	RAM	E/S	KERNAL
00110111	55	BASIC	E/S	KERNAL



Microprocesador: 6502 de MOS TECHNOLOGY de 8 bits.
Memoria: 5 Kbytes de RAM ampliables a 32 K 20 Kbytes de ROM ampliables a 28 K.
Pantalla: 23 líneas de 22 caracteres. Modulador para conectar a un televisor normal. Salida monitor video.
Colores: 8 para el marco, 16 para el fonde de la pantalla y 8 para los caracteres individuales, video inverso.
Gráficos: Semi-gráficos por teclado y alta resolución por redefinición del generador de caracteres (situándolo en RAM). Definición de 176 por 184 puntos.
Teclado: Tipo QWERTY de 62 teclas más cuatro de función definibles por el usuario.
Sonido: Tres voces de tres octavas cada una decaladas una octava entre si, resultando una extensión total de cinco octavas. Un generador de ruido aleatorio afinable para efectos especiales, un control general de volumen.
Programación: Lenguaje BASIC, intérprete residente en ROM de 8K. Posibilidad de interceptar las funciones del Basic para crear nuevas instrucciones "a medida". El Basic del Vic es uno de los rápidos actualmente en el mercado.
Complementos: Port de usuario de 8 bits entrada/salida más dos señales de sincronismo.
Bus de expansión para ampliaciones de memoria y periféricos.
Port de juegos con conexión para dos potenciómetros (paddles), y una palanca de juegos (joystick).
Almacenamiento de masa: Unidad de cassette C2N de diseño especial para registrar programas y datos.
Ampliación de memoria: En caso de ser necesario conectar más de un cartucho al mismo tiempo, está disponible un módulo (VIC 1020) que permite la conexión simultánea de hasta seis cartuchos.

VIC-1541 UNIDAD DE DISCO

Capacidad total: 174848 bytes por disco.
Secuencial: 168656 bytes por disco.
Entradas de directorio: 144 por disco.
Sectores por pista: De 17 a 21.
Bytes por sector: 256.
Pistas: 35.
Bloques: 683 (644 bloques libres).
Soportes de información: Discos standar de 5 1/4 pulgadas, de una sola cara y densidad simple.
Sistema operativo: DOS de COMMODORE inteligente (tiene procesador propio y no ocupa memoria del ordenador central).

VIC-1525 IMPRESORA

Método de impresión: Matriz de 5 × 7 puntos, impacto por un solo martillo.
Modo caracteres: Mayúsculas y minúsculas, símbolos, números y caracteres gráficos del VIC-20.
Modo gráfico: Puntos direccionables (bit image). Siete puntos verticales por columna, 480 columna máximo.
Velocidad: 30 caracteres/segundo, de izquierda a derecha, unidireccional.
Caracteres/Línea: Máximo 80. (Posibilidad de impresión en doble ancho).
Espaciado entre líneas: 6 líneas/pulgada —modo caracteres, 9 líneas/pulgadas— modo gráfico.
Alimentación de papel: Arrastre por tractor.
Ancho de papel: Entre 4,5 y 10 pulgadas.
Copias: Original más dos copias.

CARTUCHOS

Ayuda programador: Facilita la edición y depuración de programas en Basic. Instrucciones y comandos: RENUMBER, MERGE, FIND, CHANGE, DELETE, AUTO, TRACE, STEP, OFF, KEY, EDIT, PROG, DUMP, HELP y KILL.

Super expander: Intercepta el Basic del VIC permitiendo incrementar sus instrucciones y comandos en aplicaciones gráficas de sonido y juegos. Instrucciones y comandos: KEY, GRAPHIC, COLOR, POINT, REGION, DRAW, CIRCLE, PAINT, CHAR, SCNCLR, SOUND, RGE, RCOLR, RDOT, RPOT, RPEN, RJOY y RSND.
Monitor de lenguaje máquina: Facilita enormemente la depuración de programas en lenguaje máquina, es ideal como complemento del Basic para redactar y poner en marcha rutinas de alta velocidad y manejo de datos en tiempo real. Instrucciones y comandos: ASSEMBLE, BREAKPOINT, DISASSEMBLE, ENABLE, VIRTUAL ZERO PAGE, FILL MEMORY, GO, HUNT, INTERPRET, JUMP TO SUBROUTINE, LOAD, MEMORY, NUMBER, QUICK TRACE, REGISTERS, REMOVE BREACPOINTS, SAVE, TRANSFER, WALK y EXIT TO BASIC.
Además existen cartuchos de ampliación de memòria de 3,8 y 16 Kbytes.

CURSO DE INTRODUCCION AL BASIC PARTE I y II.

En forma de libro se ha editado la primera y segunda parte de un curso de Basic que parte "de cero" y está basado en el VIC-20. Van acompañados de dos cassettes con programas y ejercicios para autocontrol.

PLOTTER VIC 1520

Método de impresión: Dibujo mediante bolígrafos de diseño especial.
Color: 4 colores; negro, azul, verde y rojo con cambio desde programa.
Cabezal: Ploter X-Y tipo tambor.
Velocidad de impresión: Media de 14 car./seg.
Caracteres por línea: Máximo 80 carac., formatos de 80, 40, 20 y 10 carac./linea.
Juego de caracteres: 96.
Velocidad de dibujo:264 pasos/seg.
Longitud del paso: 0,2 mm. en dirección X e Y.
Velocidad de dibujo de línea: 52,8 mm./seg. en dirección X e Y. 73 mm./seg. en una línea a 45 grados.
Area de dibujo: 480 pasos (96 mm.) en dirección X. Programable en dirección Y (Máx. + — 999 de una sola vez).
Papel: Rollo de 4,5 pulgadas (114 mm.).

MONITOR EN COLOR C-1701

Pantalla: 13 pulgadas (330 mm.).
Capacidad de representación: 25 líneas de 40 caracteres.
Resolución: 320 líneas horizontales.
Compatibilidad: VIC-20 y COMMODORE 64.
Conectable a un registrador de video.
Amplificador y altavoz: Incorporados.







microelectrónica
y control, s.a.

Taquígrafo Serra, 7 5.° Telf. 250 51 03. BARCELONA-29
Princesa, 47 3.° G. Telf. 248 95 70. MADRID-8